



КОМПЮТЪРНО УЧИЛИЩЕ

# PC SCHOOL

ПРОЧЕТИ & РАЗБЕРИ

Брой  
**3**

вашият помощник в света на компютрите

## Photoshop

Пускане на сенки

## Flash MX

Движение по междинни фази

## 3DS Max

Depth of field във V-ray

## Word

Прибавяне на воден знак

## JavaScript

Математически операции

# Съдържание:

<b>Photoshop</b> - Четиристранен огън	<b>3</b>
<b>Photoshop</b> - Пускане на сенки	<b>4</b>
<b>Photoshop</b> - Alien ефект	<b>6</b>
<b>Draw</b> - Бутон в стил Yahoo	<b>7</b>
<b>Flash MX</b> - Движение по междинни фази	<b>8</b>
<b>Premiere</b> - Потребителски преходи	<b>11</b>
<b>3DS Max</b> - Depth of field във V-ray	<b>12</b>
<b>3DS Max</b> - Displacement във V-ray	<b>15</b>
<b>3DS Max</b> - Displacement във V-ray 2 част	<b>19</b>
<b>Word</b> - Прибавяне на воден знак	<b>22</b>
<b>Защо понякога антивирусната програма</b> не си върши работата?	<b>23</b>
<b>Безвъзвратно</b> триене	<b>25</b>
<b>PHP</b> на домашното PC	<b>29</b>
<b>JavaScript</b> - Математически операции	<b>31</b>

Автори в този брой:

**The ONE**

theone86@abv.bg

**Димитър Маринов**  
**"oorka"**

oorka@yahoo.com

**Кристиян**  
**Александров**

www.virinfo.hit.bg  
k\_alexandrov@mail.bg

Ако сте с по-стара или с по-нова версия на определена програма, недейте да пренебрегвате статиите, тъй като в повечето случаи различните версии си приличат.

Всички права върху статиите принадлежат на авторите!



# Photoshop - Четиристранен огън

(Adobe **Photoshop 6**)

Тази статия е за всички пиро-маниаци. Ще разберете как да създавате, пламтящ огнен текст.

**1.** Създайте празно изображение и го запълнете с черен цвят. Натиснете **X** от клавиатурата (за да размените цветовете) и след това щракнете върху инструмента за текст. Напишете това, което искате и натиснете **OK**.

Центрирайте текста върху изображението и след това натиснете **CTRL + E** за да го слеете със слоя на фона.



**2.** После използвайте **Filter -> Stylize -> Wind** със следните настройки: "**Wind**" и "**From the Left**". Щракнете **OK**. Сега натиснете **CTRL + F** за да приложите същия филтър повторно. След това приложете друг филтър: **Filter -> Stylize -> Wind**, но този път с тези настройки: "**Wind**" и "**From the Right**". Щракнете **OK**. Сега, както преди, натиснете отново **CTRL + F** за да приложите пак филтъра.

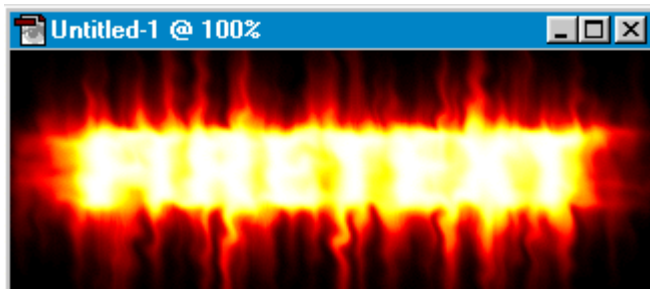


**3.** Следващата стъпка е да повторим същия процес, който приложихме на страните, но този път от горе и от долу. Тъй като в диалоговия прозорец "Wind" няма опция за горе и долу, ние ще завъртим самото изображение. Използвайте: **Image -> Rotate Canvas -> 90 CCW** Приложете вятъра точно както преди, два пъти от ляво и от дясно. Когато направите това, използвайте: **Image -> Rotate Canvas -> 90 CW** за да изправите текста.



4. Приложете: **Filter -> Blur -> Gaussian Blur** със стойност 2, после **Filter -> Distort -> Ripple** със стойности по подразбиране.

5. Конвертирайте изображението в Grayscale, като използвате: **Image -> Mode -> Grayscale** и после в **Image -> Mode -> Indexed Color**. Следващата стъпка е да зададете някакъв цвят на изображението, така че изберете: **Image -> Mode -> Color Table**. Както знаете, ние искаме "огнен ефект", изберете "**Black Body**" раздела.



6. Последна стъпка: направете текста ни малко по-четлив. Изберете **Text Tool** и натиснете **D** от клавиатурата. Използвайте/въведете същите думи, от по-рано (същия размер на шрифта и вид), щракнете **OK**, позиционирайте текста в средата и кажете "Уау!" :).



Превод: The ONE    Източник: [www.absolutecross.com](http://www.absolutecross.com)

## Photoshop - Пускане на сенки

(Adobe Photoshop 6)

Тази статия, ще ви покаже няколко начина за пускане на сенки в Photoshop. Въпреки, че тази техника е много елементарна, ще разберете, че ще я използвате много често във вашата работа.

В по-новите версии на Photoshop, има вграден начин за пускане на сенки, под вашия текст или обект: **Layer -> Layer Style -> Drop Shadow**.

Понякога това не дава ефекта, който търсите, така че ако сте упорит или използвате по-стара версия на Photoshop, давайте напред...

1. За да започнете, ще имате нужда от нещо, към което да прибавите сенки. Това може да бъде: текст, лого, картинка... Ще трябва да бъде в собствен отделен слой. Натиснете "**D**" от клавиатурата, за да върнете цветовете към стойностите им по подразбиране (ние ще използваме черно).

2. Ако ще поставяте сенки на текст, изберете **Type Tool** и въведете желаното. Направете нов слой и го дръпнете под слоя на текста ви... може да го именувате "**Shadow**" (сянка). Задръжте **CTRL** от клавиатурата и щракнете на слоя с текста, за да се се-

лектира около него.

3. Сега изберете **Select -> Feather** въведете число и натиснато **OK**. Колкото по-високо е числото, толкова по "мека" и разстлана ще е сянката. В примера е използвана стойност 5. Уверете се, че слойът на сянката е избран и отидете на **Edit -> Fill -> Foreground Color**, за да запълните селекцията с черно.



4. В зависимост от посоката на "светлинния източник" (върхова лява част, долна дясна и т.н.), трябва да преместите слоя със сянката нагоре/надолу и вдясно/наляво. В примера е преместено 5 надолу и 5 вдясно. Понякога може да създавате илюзия, че обекта се намира високо, като използвате по-високи числа... и забележете, че числата не трябва да са еднакви (т.е. може да са 4 и 7, и т. н.).



5. Вероятно забелязахте, че сянката изглежда малко нереалистично тъмна. За да оправите това, нагласете непрозрачността на слоя със сянката - в примера, е намалена до 60%.



# Dropshadow

Превод: The ONE    Източник: [www.absolutecross.com](http://www.absolutecross.com)

## Photoshop - Alien ефект

(Adobe Photoshop 6)

1. Създайте нова RGB картинка с която да работите. Примера е 325 x 125, резолюция 72 и черен фон.
2. Изберете ярко зелен цвят за преден план и черно за фон.
3. Изберете **Edit -> Fill -> Background Color**, за да запълните фоновия слой с черно.
4. Селектирайте **Type Tool** и изберете **Type Mask Tool** от Options барът в горната част на екрана (в по-ранни версии, задръжте надясно иконата на Type Tool). После щракнете в документа и напишете какъвто и да е текст, по ваш избор (изберете какъвто и да е шрифт или размер). Щракнете отметката в барът за настройки, за да завършите текста (ОК в по-стари версии).

Уточнение: Type Tool – Тоест инструмента за текст. Има буква T на него.

5. Изберете **Edit -> Fill -> Foreground Color** (зеленият).



6. Отидете на **Select -> Feather 2** пиксела.
7. Изберете **Filter -> Blur -> Radial Blur** (100, Zoom, Best).
8. Без да деселектирате, повторете стъпка 7 (натиснете **CTRL + F**).



9. Сега дайте **Select -> Inverse** и отново повторете стъпка 7... **Filter -> Blur -> Radial Blur**

(100, Zoom, Best).



Превод: The ONE    Източник: [www.absolutecross.com](http://www.absolutecross.com)

## Draw - Бутон в стил Yahoo

(Corel Draw 7)

Тук ще разберете как да създавате бутони в стил Yahoo. Много хора ми писаха въпроси, как са направени и аз ще демонстрирам тази техника. Използвах Corel Draw 7 за Windows. Някои неща може да са различни в другите версии.

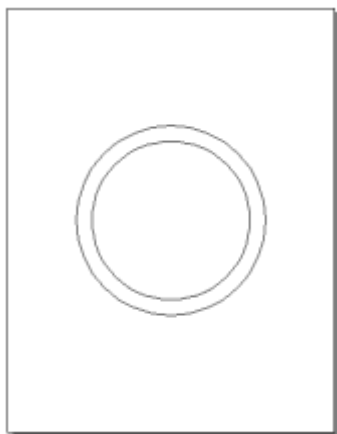
Понякога е по-трудно да направите нещо видяно в програма за чертане, вместо в програма за рисуване. Това не е такъв случай, но както и да е. Всъщност, ако нещата са направени с програма за чертане, може лесно да прибавяте отличителни черти, като например текст увит около бутон.

Започнете с начертаването на кръг. Задръжте **CTRL** докато влачите курсора, за да се уверите в кръглата форма вместо в овалната.

Повторете процеса, но този път с по-малък кръг.

Използвайте Pick инструмента за да изберете двата кръга.

Изберете: **Arrange, Align and Distribute** и в диалоговия прозорец "**Align and Distribute**" кликнете върху "**Center of Page**". Щракнете **OK**. Трябва да получите нещо подобно като на фигура 1.

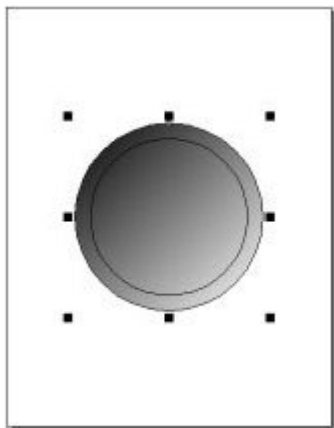


(Фиг. 1)

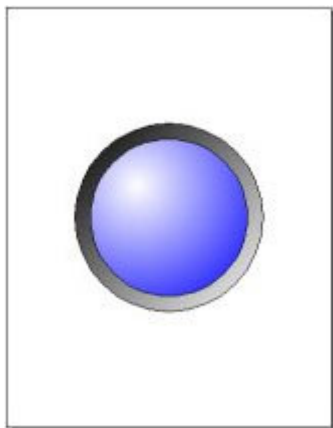
Изберете Pick инструмента и щракнете върху външния кръг, за да го селектирате.

Поставете мишката над инструмента **Fill** и задръжте левия бутон (на мишката), докато се появи плаващото меню. Изберете **Fountain Fill** иконата. Това ще отвори диалоговия прозорец "**Fountain fill**".

Поставете **Type** на **Linear** и **Angle** към **-45%**. За **Color Blend** позволете **2** цвята (Two color), (**From:** трябва да бъде черно, а **To:** бяло). Щракнете **OK** и изображението ви, трябва да прилича на фигура 2.



(Фиг. 2)



(Фиг. 3)



(Фиг. 4)

Изберете вътрешния кръг.

Отидете пак до диалоговия прозорец "**Fountain fill**". Този път поставете **Type** на **Radial**, за **Horizontal Offset** дайте **-20** и на **Vertical Offset** **20**.

Щракнете върху цвета на **From**. Ще се отвори малко меню за цветове. Изберете цвета в който искате да бъде бутон. Аз избрах тъмно син. Щракнете **OK** и трябва да получите нещо като фигура 3.

Това е за бутон. Сега може да прибавите текст или изображение. Аз добавих Yippee! икона и поставих текст свит надолу в кръг. Текста е разположен в горния край. После запаметих картинката в JPG, може да я видите на фигура 4.

Уверете се, че сте съхранили файла в CDR, така ще може по-късно да се връщате и да правите промени.

Това е! Приятен дизайн.

**Превод:** The ONE **Източник:** [www.grafx-design.com](http://www.grafx-design.com)

## Flash MX - Движение по междинни фази

(Macromedia **Flash MX 2004**)

Какво точно е движението по междинни фази? Първо, нека да си представим, че сте поставили кутия върху маса. Сега си представете, че след 10 секунди, кутията неочаквано изчезва и магично се появява отново, но на няколко инчове вдясно от там, където беше оригинала. Доста странно нали?

Добре, ако можехте да върнете назад миналата минута и да я прегледате отново, какво щяхте да видите? Ще видите кутията неочаквано да изчезва и да се появява отново в различна позиция с никаква последователност или "анимация" в 10 -те секунди, които изтекоха. Ето там е мястото, където трябва да се вмъкне движение по междинни фази. Така вместо кутията да изчезва и да се появява, ще видите как тя бавно се мести към новата си позиция - няма никакво изчезване! Междинните фази, просто запълват липсващия преход, между началната и крайната позиция. Така се създава пълна анимация.

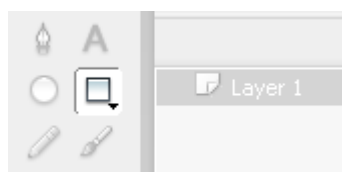


Тази статия, обяснява начина, по който се създава движение по междинни фази във Flash MX 2004 и за щастие ще разберете повече за самото движение.

### Научете как се прави движение по междинни фази:

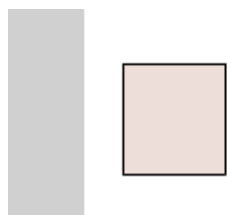
**1.** Първо, пуснете Flash MX 2004. След като програмата се зареди, трябва да виждате, нова празна област - наречена работно пространство. Ако не виждате каквото и да е наподобяващи бяла площ, отидете на: File -> New -> Flash Document.

**2.** Сега, щракнете на инструмента "Rectangle" от иконите вляво. Иконата на инструмента е показана на следващото изображение:



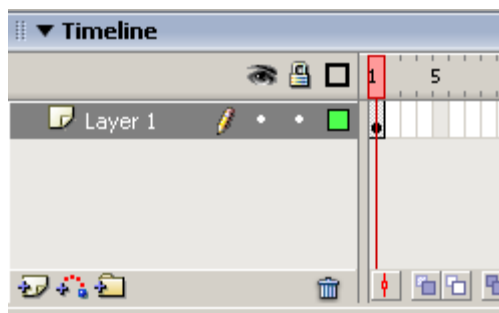
(щракнете върху иконата за правоъгълник)

**3.** След като щракнахте иконата, отидете към вашата работна област и нарисувайте квадрат, щраквайки с мишката, като я влачите и накрая отпуснете бутона. Уверете се, че квадрата ви е близо до левият ръб на работната област:



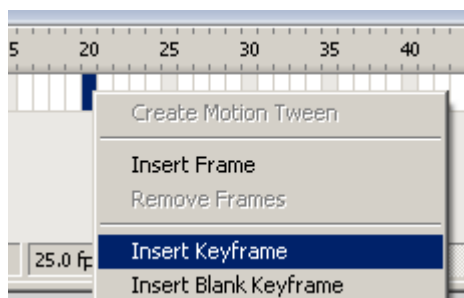
(нарисувайте квадрат - уверете се, че е подравнен към лявата част)

**4.** След като квадрата е нарисуван, погледнете в панела "Timeline". Timeline е голяма област от числа с малки вертикални кутийки. Прието е, че при кадър 1, се появява запълнена точка:



(забележете пълната точка в "Timeline")

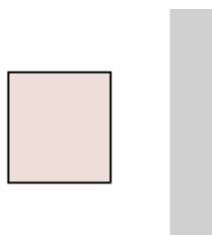
**5.** Пълната точка означава, че има нещо съществено в този кадър. В нашия случай, това е квадратът, който създадохме! Сега, погледнете в дясно и намерете числото 20. Щракнете десния бутон в кутийката, под него и изберете "Insert Keyframe":



(изберете Insert Keyframe в кадър 20)

**6.** След като вмъкнете ключов кадър (Keyframe), ще видите друга черна точка в кадър 20. Не се безпокойте, така трябва! Сега се уверете, че сте избрали кадър 20, като щракнете на него с курсора на мишката.

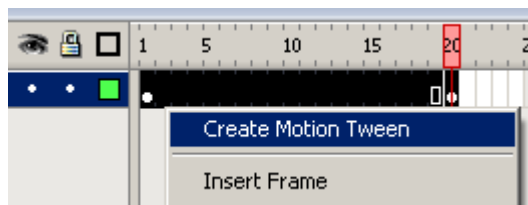
**7.** След като изберете кадъра, отидете отново при вашия квадрат. Забележете, че квадрата също е избран. Сега, натиснете дясната стрелка (от клавиатурата) няколко пъти, за да достигнете на няколко сантиметра от дясната страна на работната площ:



(квадрата е преместен в дясно)

**8.** Сега с курсора на мишката щракнете в кадър 1 на "Timeline". Забележете къде се намира квадратът! Той е в ляво. Без да се колебае, сега щракнете в кадър 20. Квадратът е в дясно. Щракнете някъде другаде, например в кадър 10, забележете как квадрата се върна обратно в оригиналната си позиция. Тоест в ляво.

**9.** Време е, за движение по междинни фази! Щракнете в кадър 1 и селектирайте всичко до 20 -ти кадър (като задържите левия бутон). Кутийките трябва да станат черни, щракнете с десния бутон и изберете "Create Motion Tween":



(селектирайте всички кадри и изберете "Create Motion Tween")

**10.** Ако всичко е както трябва, кутийките ще приемат лилав цвят със стрела от 1 -ви към 20 -ти кадър. Щракнете където и да е върху "Timeline" и проследете позицията на квадрата. Вижете как се движи.

**11.** Отидете на File -> Save - за да запишете анимацията. После изберете File -> Publish Preview -> HTML. Трябва да се появи квадрата ви, който ще се мести постепенно от ляво на

ДЯСНО.

Надявам се, че тази статия ви е помогнала да разберете, какво е движение по междинни фази. Припомнете си примера за кутията, който използвах по-рано. Забележете, че преди прехода, кутията просто изчезваше и се появяваше в нова позиция. Подобно на това си спомнете, че квадрата между кадър 1 и 20 рязко преминаваше, а с прилагането на преход (междинни фази) се получи плавност.

Междинните фази промениха позицията на квадрата или кутията, при тяхното изпълнение. И то доста реалистично и "живо". Стига вече със случайното изчезване и появяване.

**Превод:** The ONE **Източник:** [www.kirupa.com](http://www.kirupa.com)

# Premiere - Потребителски преходи

(Adobe **Premiere 5.1**)

В тази статия, ще изследваме в дълбочина преходите на Premiere 5.1. Ще научите как да пригаждате преходи за да създавате заслепяващи ефекти. Също ще разгледаме и филтрите.

Premiere 5.1 дава няколко алтернативи, за прилагане на потребителски преход, като позволява да запаметите прехода си и да го използвате по-късно.

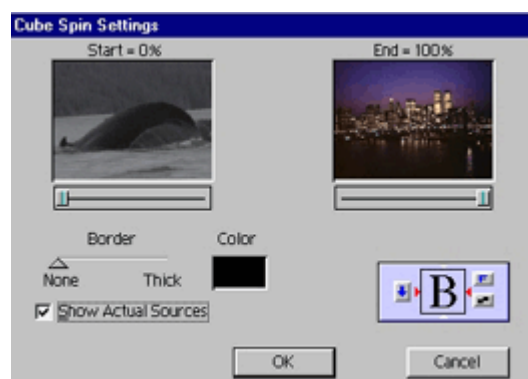
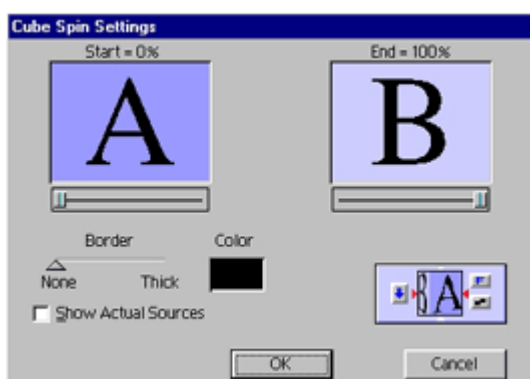
За да прибавите преход към проекта ви, вашите клипове трябва да бъдат на видео пътеките 1A или 1B.

## Потребителски преходи

Premiere има над 70 прехода, но повечето с прости конфигурации. Ако прибавите малко съзидателност, ще имате безкрайни възможности.

## Използване на диалоговия прозорец "Настройки на преходи" (Transitions Settings)

Пуснете прехода в пътеката за преход (Transition Track) и щракнете двойно на него, за да се появи диалоговия прозорец "Transitions Settings". Щракнете отметката **Show Actual Sources**, за да се покаже актуалния източник.

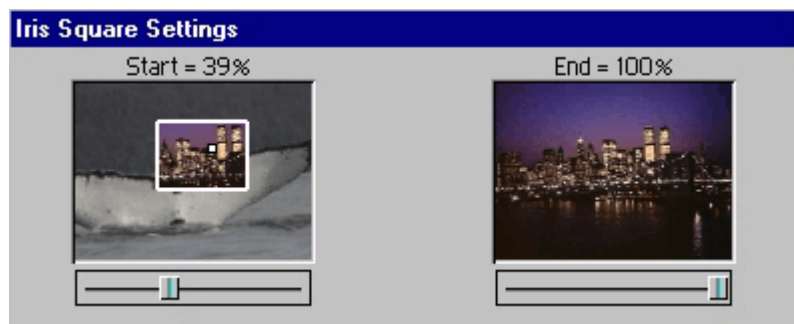


## Настройка на начална и крайна точка

Променяйки началната и крайната точка, може да започнете и прекратите прехода в средата на ефекта. Съвет: задръжте долна стрелка (от клавиатурата), за едновременно заключване и

премествайте началния и крайния плъзгач.

Когато използвате преходи за: Zoom, Iris Cross, Iris Diamond, Iris Round, Iris Square или Iris Star, може да промените началната позиция. Просто преместете курсора на мишката, за разместване.



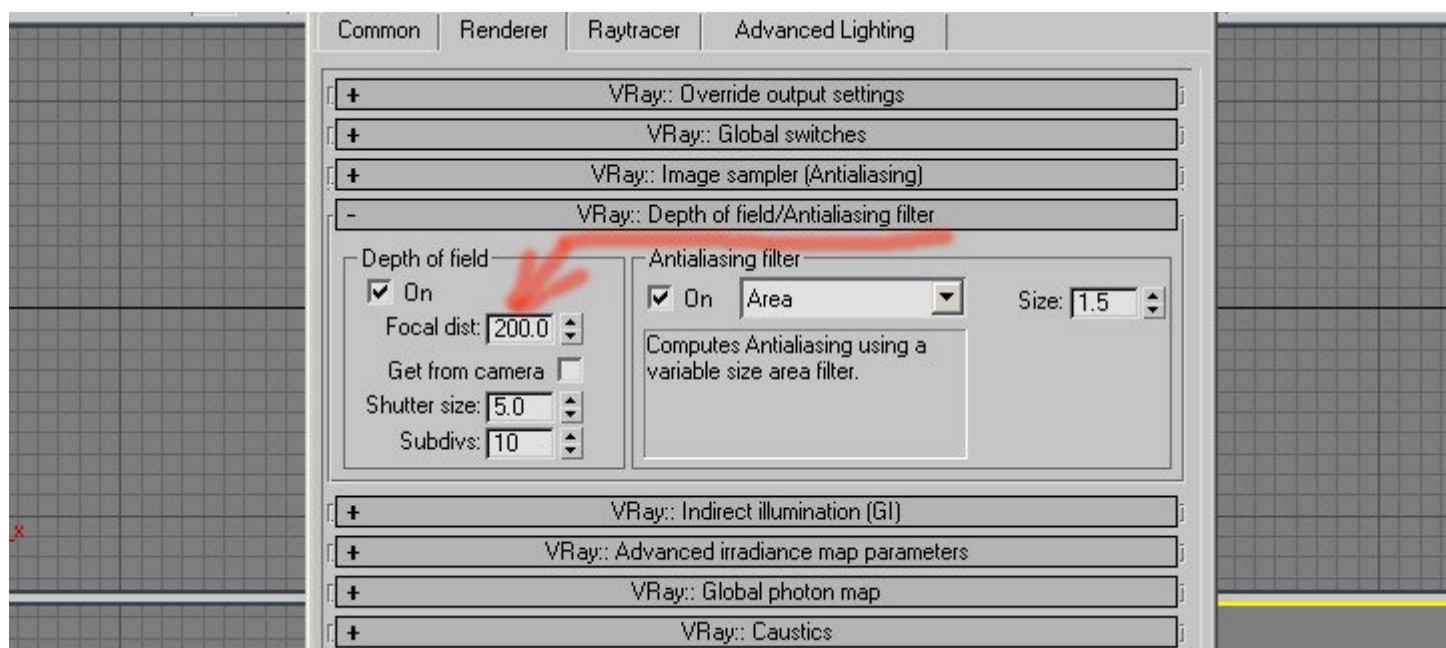
Също така, може да прибавите ръб (граница) около прехода. Използвайте **border** плъзгача, за да нагласите плътността и **color picker**, за да промените цвета на границата.

**Превод:** The ONE **Източник:** [www.digitaleditor.com](http://www.digitaleditor.com)

## 3DS Max - Depth of field във V-ray

(Discreet **3DS Max 6**)

**Depth of field** - ефект позволяващ визуализацията на дадено изображение по същия начин, по който би изглеждало ако е заснето от камера, фокусирана на определена точка от сцената.

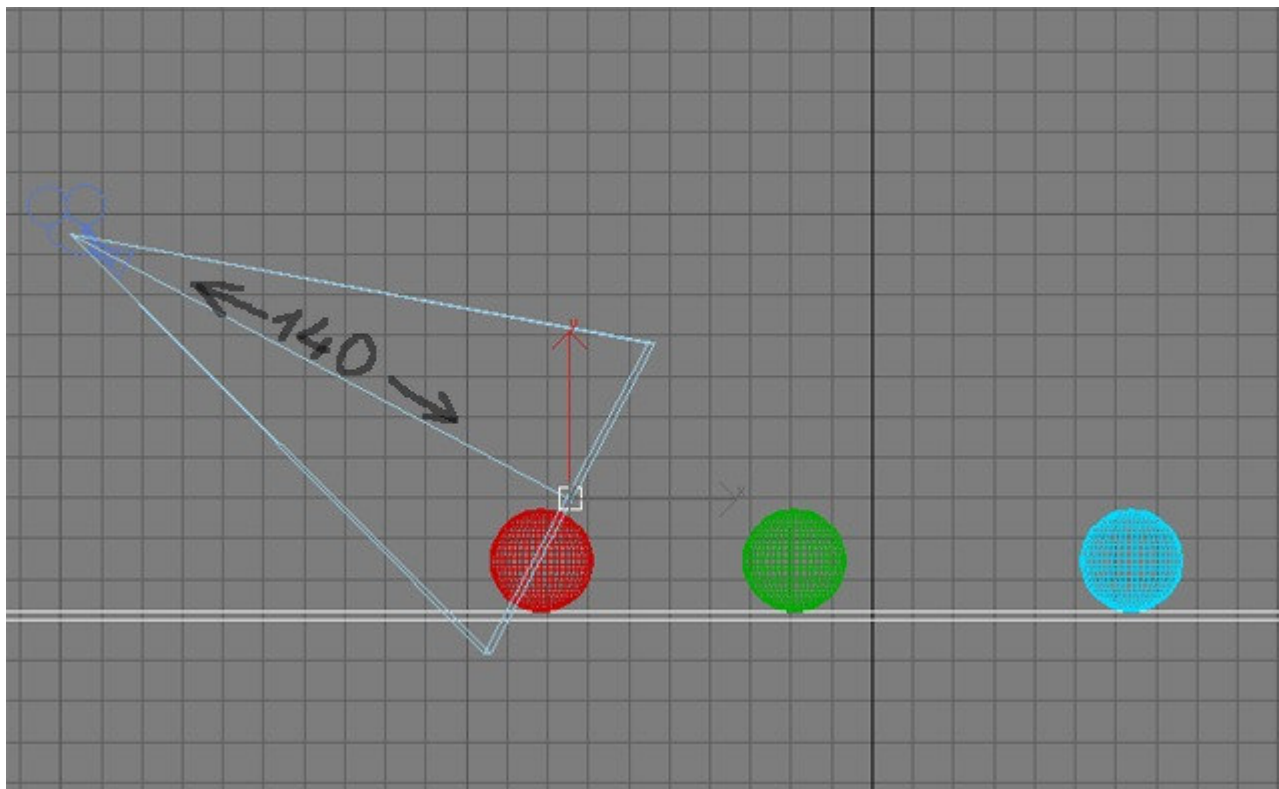


Да разгледаме основните параметри на ефекта:

**Focal dist** - Разстоянието от гледната точка до обекта, в който е фокусирана камерата.



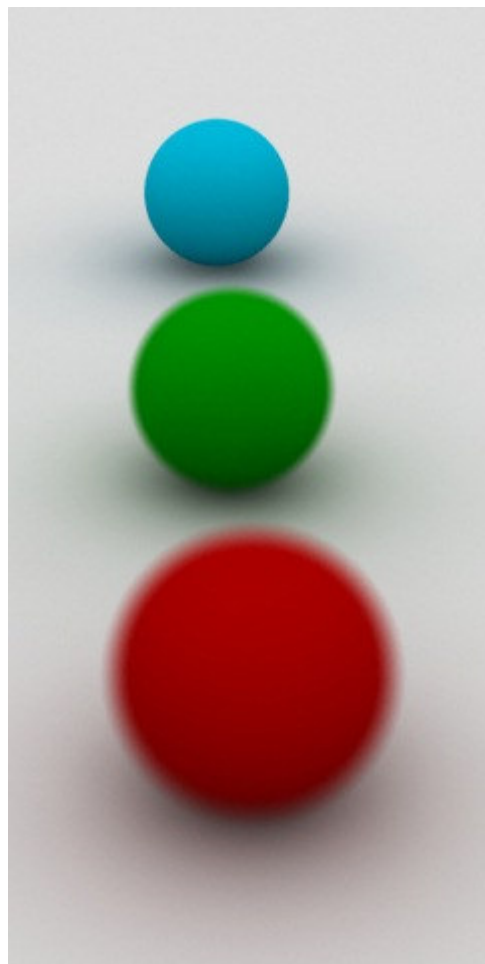
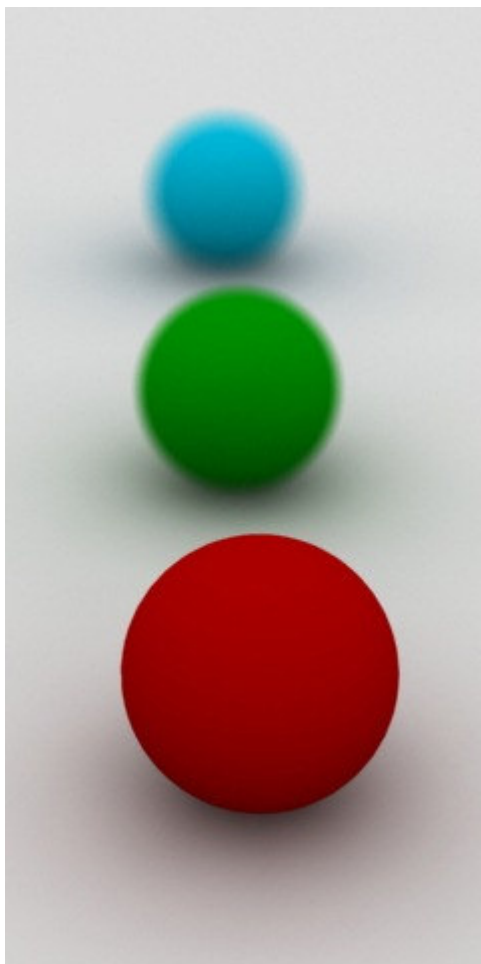
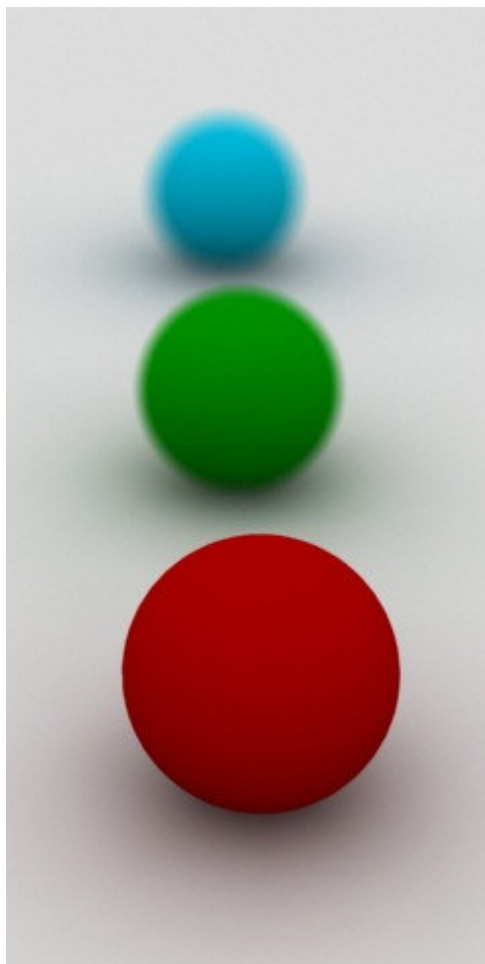
**Get from camera** - когато тази опция е включена, фокусното разстояние ще бъде взето автоматично от камерата или това е разстоянието от **Camera01** до **Camera01.Target**



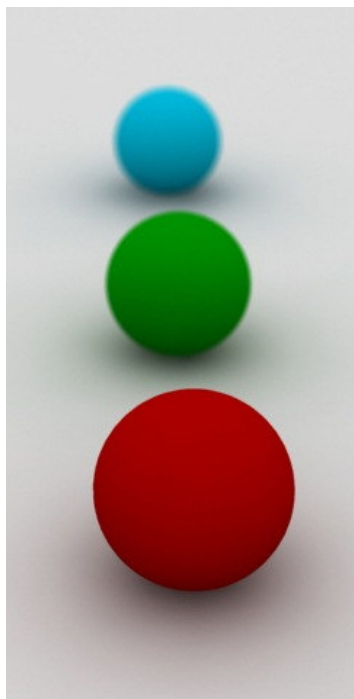
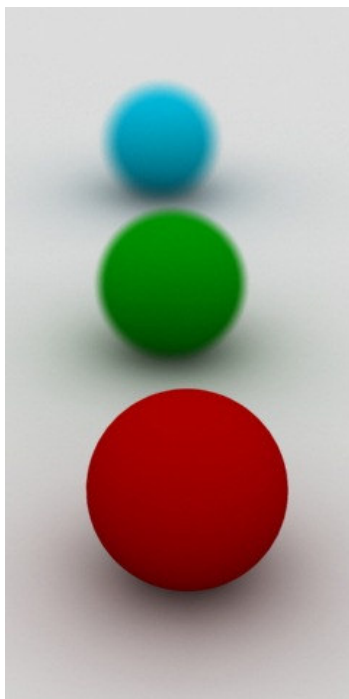
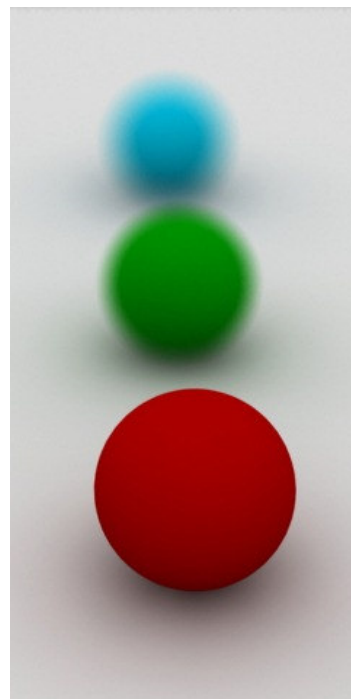
Get from camera: **on**

Focal dist=**140**

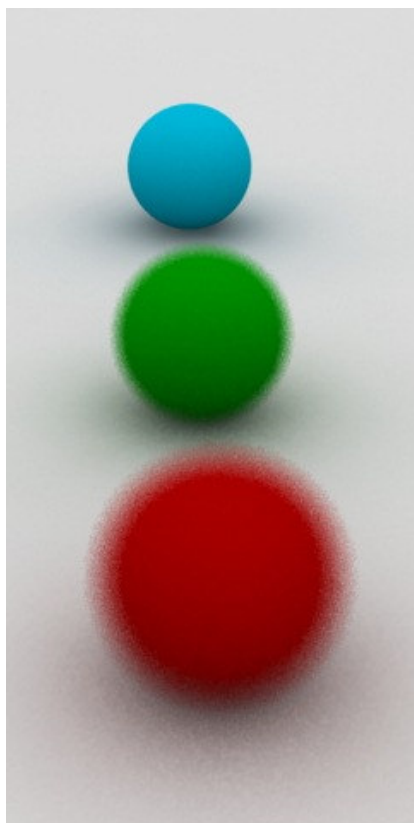
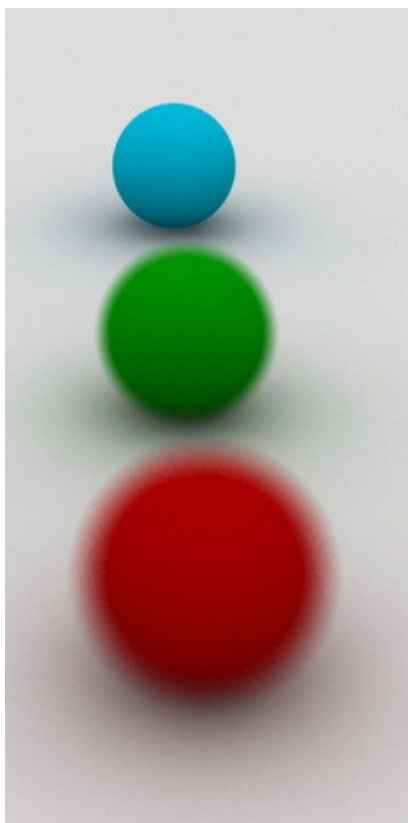
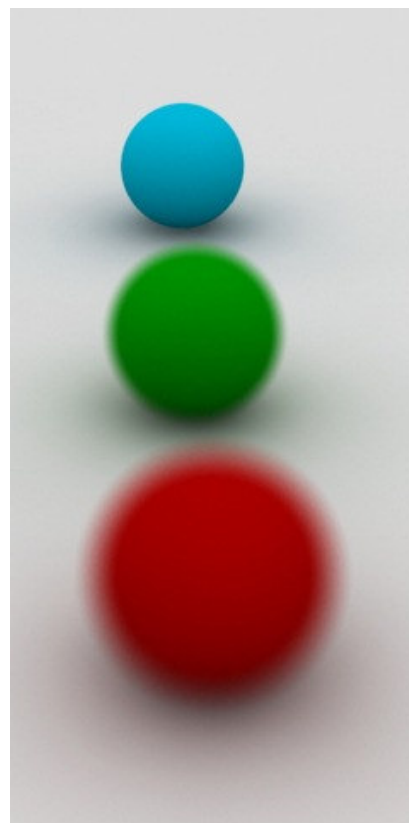
Focal dist=**270**



**Shutter size** - Стойност, от която зависи степента на "разфокусиране". По-големите стойности предизвикат по-голямо "разфокусиране"

**Shutter size = 1.5****Shutter size = 3.0****Shutter size = 6.0**

**Subdivs** - Стойност, определяща броя семпли. По-високи стойности дават по-качествен резултат. Съответно времето за визуализация се увеличава

**Subdivs = 2****Subdivs = 8****Subdivs = 20****Rendertime = 00:46****Rendertime = 02:44****Rendertime = 06:38**

Пример за сцена с **Depth of field (DOF)** ефект:



Автор: Димитър Маринов "oorka"

## 3DS Max - Displacement във V-ray

(Discreet 3DS Max 6)

**Displacement** - изместване на повърхността на обекта. Намира широко приложение при моделиране на повтарящи се елементи. Това става с помощта на чернобяла карта, където белият цвят определя участъците с максимално изместване, а черният цвят участъците без изместване...

В този кратък урок ще разгледаме модификатора **Displacement** във визуализатора **V-ray**. Заредете **3ds MAX** и създайте **plane** с произволни размери, примерно близки до квадратна форма. След това от **modify** прибавете модификатора **UVW mapping->planar**. Следващата стъпка е прибавянето на модификатора **VRayDisplacementMod**. Важно е да се запомни, че преди прибавянето на модификатора **VRayDisplacementMod** трябва да се приложи задължително модификатор за координатна проекция **UVW mapping**. В противен случай няма да се получи желанния резултат.

Да разгледаме основните параметри на модификатора **VRayDisplacementMod**:

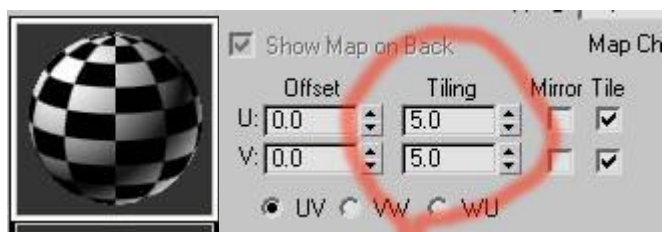
**Texmap** - карта на отместването. При натискането на този бутон се отваря прозореца на

**Material Map Browser**, и тук се указва съответната карта (задължително черно-бяла)

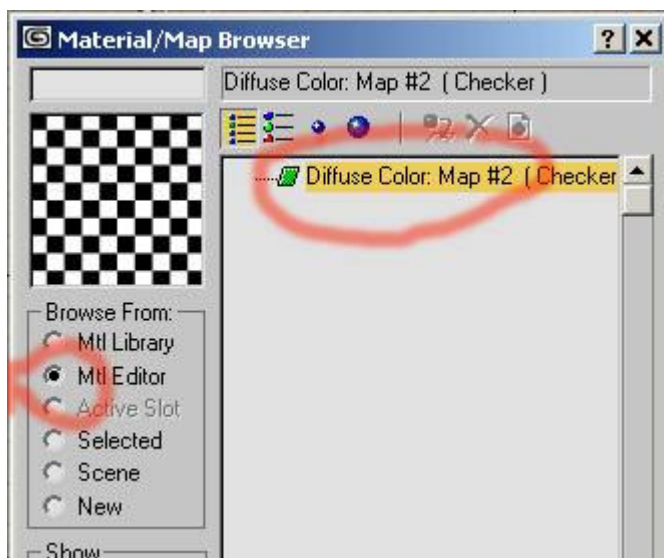
**Amount** - величина, определяща големината на изместване на повърхността на обекта

**Shift** - величина, повдигаща целия обект

**Precision** - влияе върху точността на изчисленията

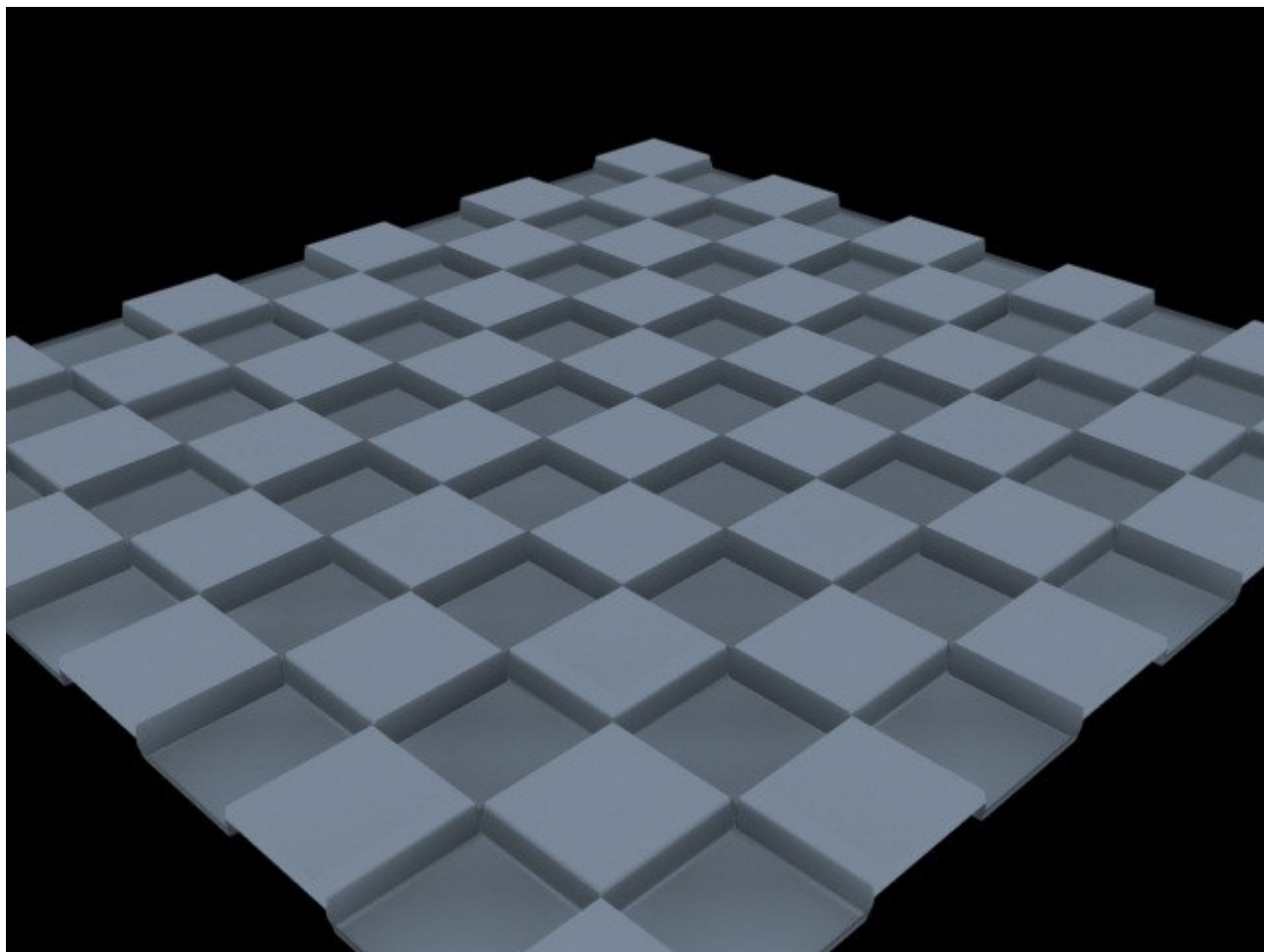


В **Material Editor** съм заредил карта тип checker, като tiling е със стойности: 5,0 и 5,0



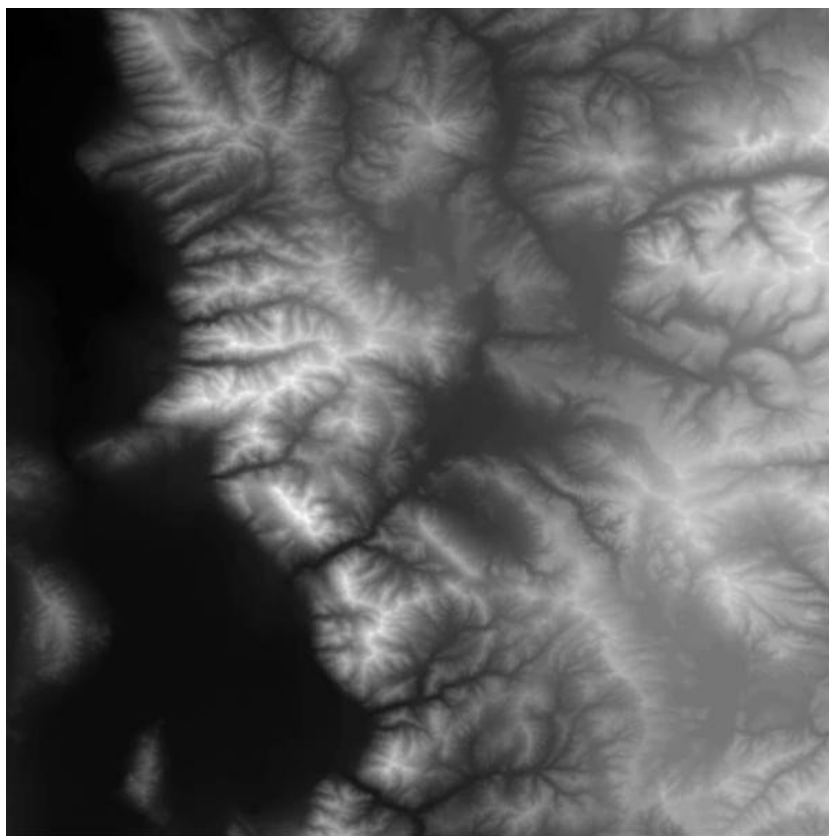
Натиснете бутона **Texmap** и от **Material/Map Browser** заредете току що създадената карта

Ето резултата след визуализацията (стойността на Amount е 5.0 в този случай):

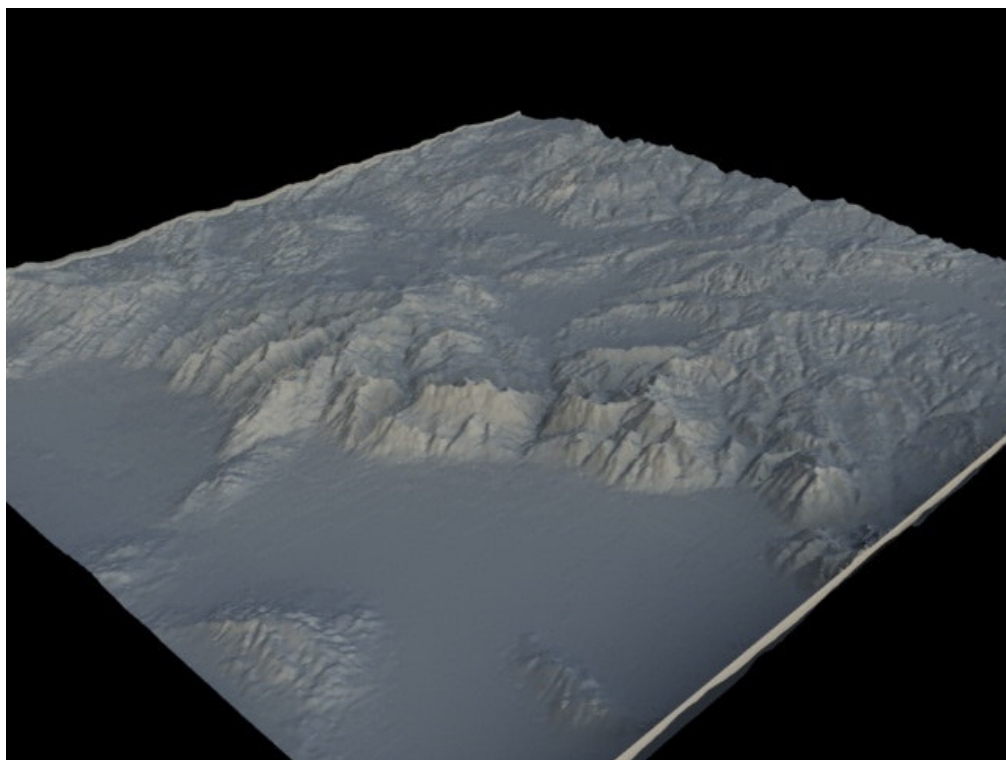


В следващия свободен слот на **Material Editor** съм заредил едно bitmap изображение:

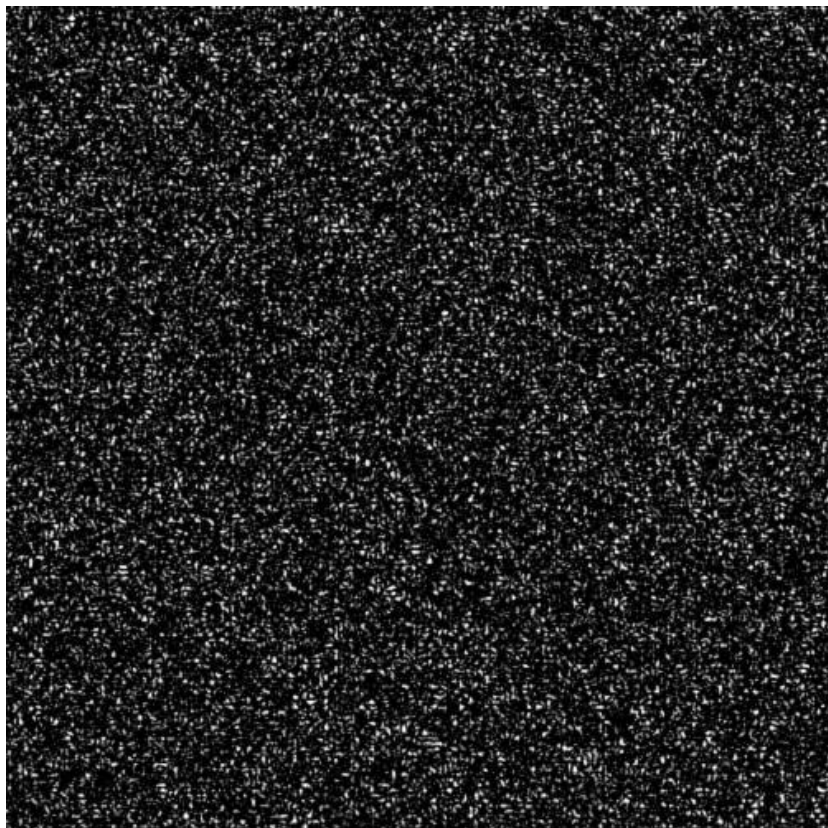




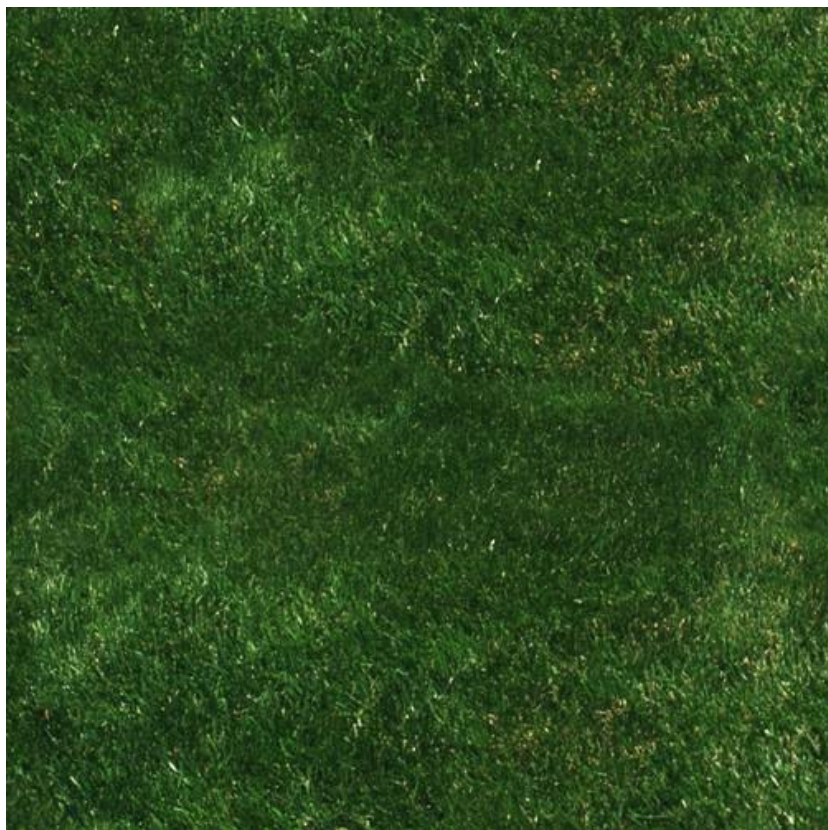
Това е черно бяла карта на долината(???) Солтлейк, Америка. Белият цвят обозначава по-високите точки, съответно черният - по-ниските. Натиснете бутона **Texmap** и от **Material/Map Browser** заредете тази карта. Ето резултата след визуализацията (rendering):



Сега ще разгледаме един елементарен пример за създаване на **трева**. Създадох следващото изображение от няколко **Noise (turbulence)**.

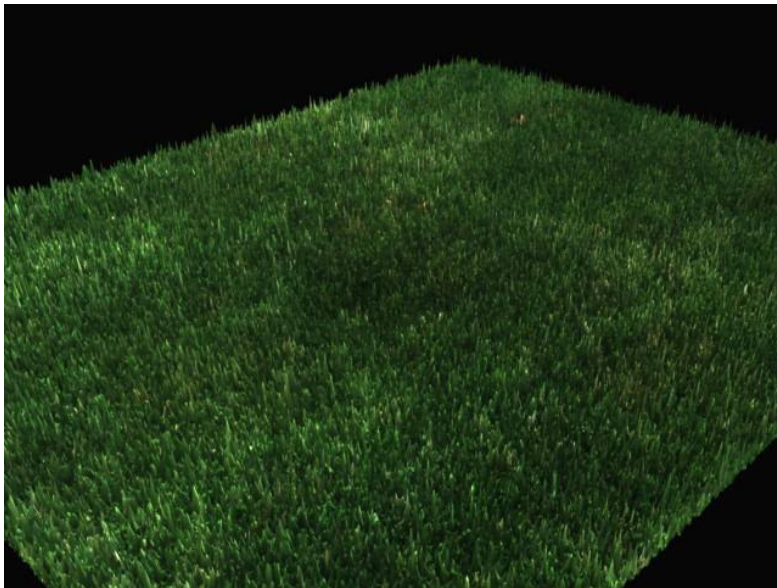


Към самия **plane** приложих първата ми попаднала текстура на трева::



Ето резултата след визуализация (rendering):



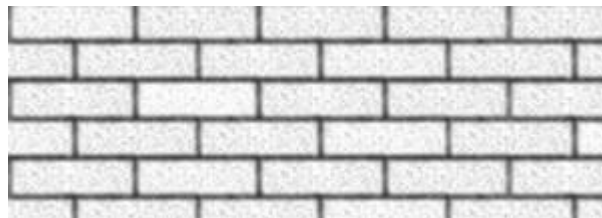
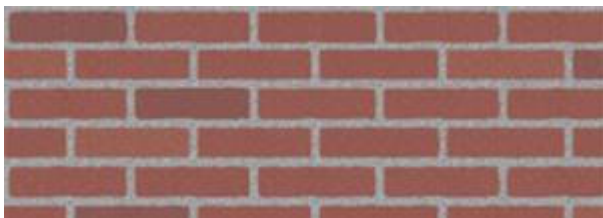


Автор: Димитър Маринов "oorka"

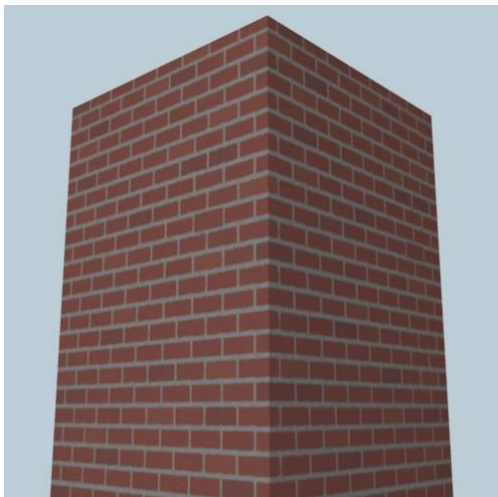
## 3DS Max - Displacement във V-ray 2 част

(Discreet 3DS Max 6)

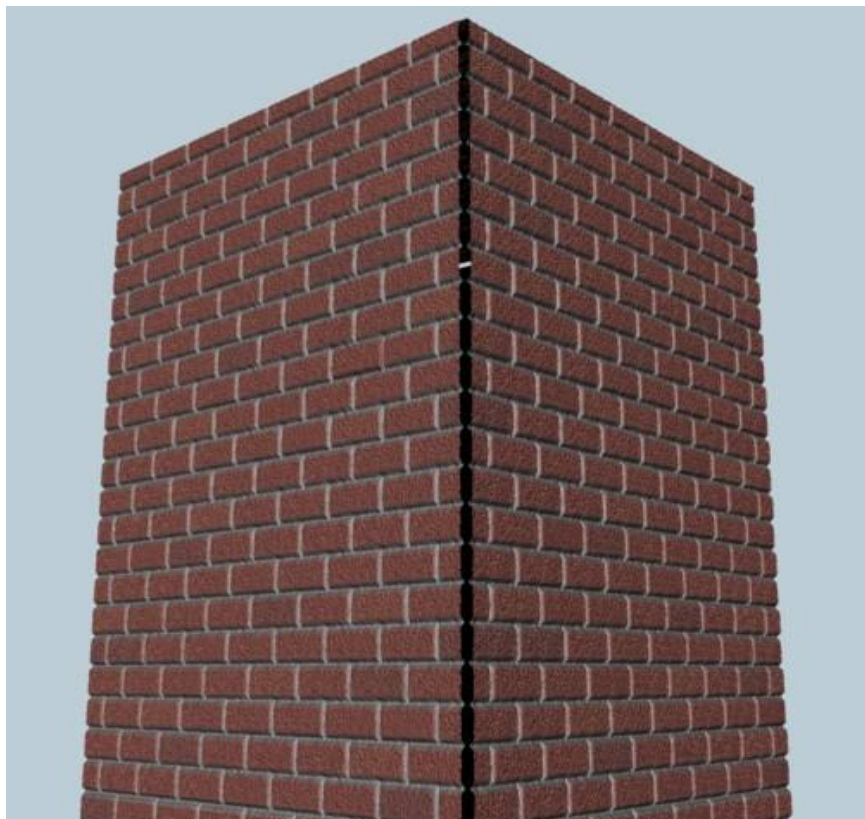
Тук ще разгледаме един малък трик, за избягване на разпокъсванията, които се явяват по ръбовете на обекта при ползване на модификатора **VRayDisplacementMod**. За целта съм създал проста сцена, състояща се от **box**, светлинен източник тип **omni** и **камера**. В редактора за материали в първи и втория слот в **maps/DiffuseColor** съм заредил две изображения:



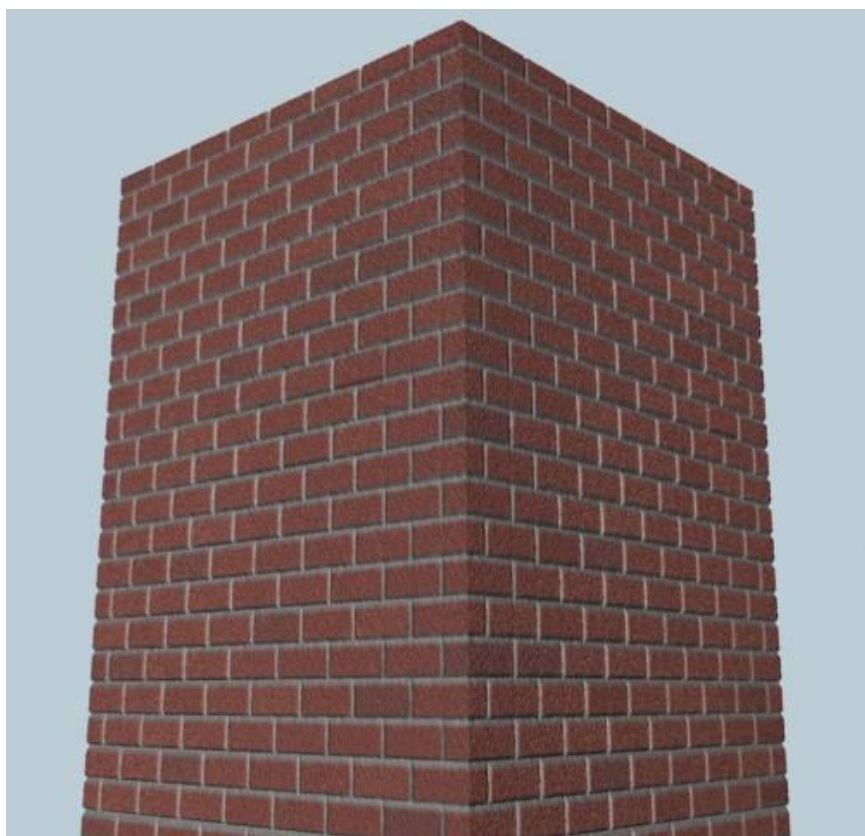
Първото изображение съм го присъединил върху обекта **box01** в сцената. Съответно съм приложил модификатор **UVW mapping -> box**, и съм нагласил материала както желая. Дотук резултата от нашите действия е следния:



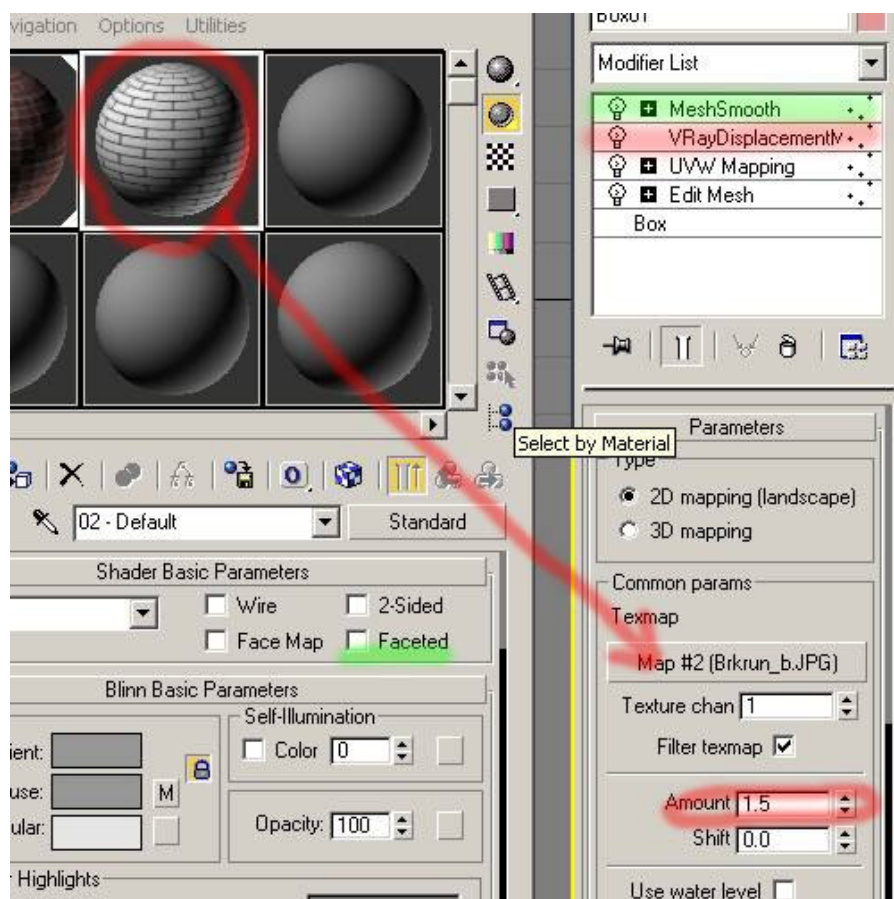
Следващата стъпка е да приложим модификатора **VRayDisplacementMod**. В **texmap** зареждаме второто изображение от редактора за материали, което всъщност е черно бяло копие на първото и ще ни служи за карта на изместването. Amount=1.5 . Визуализираме



Остава да се "разправим" с досадните разкъсвания по ръбовете на обекта. За целта добавяме модификатор **MeshSmooth** върху обекта **box01** и оставяме броя на итерациите равен на 0. След това в редактора за материали даваме отметка **faceted** за материала на обекта **box01**. Визуализираме:







Тази техника съм използвал за комина и каменните стени на къщата в следващата сцена:



**Автор:** Димитър Маринов "oorka"

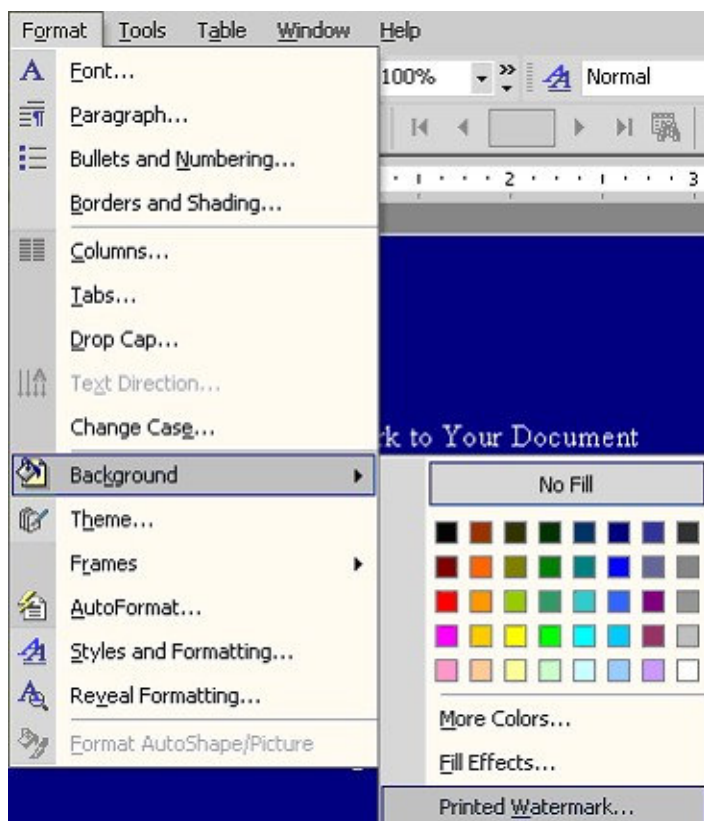
# Word - Прибавяне на воден знак

(Microsoft **Word 2002**)

Водния знак може да бъде ценно допълнение към документа ви. Може да подобрите външния вид на документа, като прибавите клеймо, изображение или пък воден знак на текст. Така ще може да идентифицирате съдържанието на документа като чернова, поверителна информация или нещо друго. Отличителната черта на водният знак в Word е определен актив за средни до напреднали потребители на Word.

За да прибавите воден знак към документа ви, той трябва да е показан в **Print Layout** или **Normal View**.

**1.** От менюто **Format**, изберете **Background** и след това **Printed Watermark**.



**2.** В диалоговия прозорец **Printed Watermark**, щракнете радио бутона преди **Text Watermark**.



3. В падащото меню **Text**, изберете водния знак, който желаете.
4. Използвайте и другите падащи менюта: **шрифт**, **размер**, **цвят**, за да форматираме водния знак.
5. Уверете се, че има отметка пред **Semitransparent**, за да може водния знак да не взаимодейства с текста на документа.
6. Може да промените ориентацията на текста от **Layout**, избирайки диагонална или хоризонтална подредба.
7. След като изберете водна марка и приложите необходимите настройки за форматиране, натиснете **OK**.

Водния знак, ще бъде приложен към документа ви. Може да го видите на екрана и когато отпечатвате документа.

Превод: The ONE    Източник: [www.tutorialized.com](http://www.tutorialized.com)

## ЗАЩО ПОНЯКОГА АНТИВИРУСНАТА ПРОГРАМА НЕ СИ ВЪРШИ РАБОТАТА?

Надявам се това обяснение да ви е полезно и да сложи край на безкрайното питане защо антивирусна не прави това или онова въпреки че производителя твърди че програмата го може и го прави!

Всяка операционна система работи по различен начин. Антивирусната програма работи различно под всяка операционна система. Антивирусната програма и операционна система си влияят взаимно една на друга. Това е причина, проблемът може да е поради особеност на операционна система, а не поради бърз на антивирусната програма. Различните настройки на операционна система карат операционна система да се държи различно. Различните настройки на антивирусната програма карат антивирусната програма да се държи различно. Те взаимно си влияят и си влияят различно, според това с коя операционна система си и как е настроена с коя антивирусна програма си и как е настроена антивирусната програма. Понякога антивирусната програма не може да си свърши работата не защото тя самата не може, а защото операционна система ангажирайки определен ресурс, ангажира и файла който отговаря за този ресурс, съответно антивирусна програма не може да го ползва, защото той е зает и е принудена да чака докато операционна система го освободи. принудително освобождаване на ресурсите под Windows става като се пусне Windows в Safe Mode. Тогава със свободни ресурси, антивирусната програма може да действа свободно и спокойно, защото няма кой да и се бърка в работите. Всяка антивирусна програма, разчита, на операционната система, да я стартира на определен етап от зареждането на системата. когато windows тръгва в Safe Mode, просто самият windows умишлено не стартира нищо друго освен себе си. няма Direct X, няма работещи services няма резидентен антивирус, няма драйвери, няма локална мрежа или интернет, умишлено е направено така, за да може всичко като ресурси да се освободи. антивирусната програма също не се стартира, и не е бърза а просто никой не и казва - ей стартирай се, че трябва да ми пазиш компютъра. тогава ръчно пускате антивирусната програма да сканира диска, и съответно да чисти. вече антивируса тръгва да работи, и да чисти защото вие сте му казали - ей, стартирай се, че трябва да ми изчистиш компютъра. Колкото повече функции, шарени има една операционна система, plug and play, съветници, Wizards, ани-

мации, и други такива глезотиинки като шарени ленти, шарени бутони, 3D ефекти на бутоните - както е при Windows XP, то Операционната система взема повече ресурси и съответно файловете отговарящи за тези ресурси. Това значително увеличава шанса, антивирусната програма да поиска вече зает ресурс, и ако антивирусната програма не получи ресурса, ще каже ама не мога да го направя защото еди какво си ми пречи или е заето. Антивирусната програма сама си казва точно какъв е проблема на който се натъква. Последното нещо, което добре да знаете, е че, Антивирусната програма има за цел да спре вируса да не ви зарази, и да го премахне от заразената система, но антивирусната програма не може да поправи повредите нанесени от сериозен вирус и/или ако вирусът е работил дълго време на компютъра. Антивирусната програма само спира вируса да не влезне, или го намира къде се е скрил, ако вече е влезнал и го премахва. Нищо повече от това.

Производителя на всяка операционна система и производителя на всяка антивирусна програма, за съответната операционна система, прилагат 3 файла с информация за продукта:

Четейки документацията на антивирусната програма вие научавате нейните особености как работи, защо така, и работата ви с нея става по-ефективна.

**quick guide** - брошура - тя дава обща информация за продукта - за какво е предвиждан и правен продукта, какви изисквания има, какви части/модули има продукта и общо описание кое какво прави.

**User's guide** - пълно ръководство на продукта, пълно и подробно описание на опциите, частите/модулите са описани подробно, кое как работи, защо така, кои опции какво правят, как го постигат, защо производителя е избрал този подход, какво е искал да постигне.

**help information** - помощна информация - там има описание на проблеми, и там си пише какво се прави, как се прави за да ти реши проблема, пише защо така, и ти дава и алтернативен вариант в случай че по някаква причина с основния вариант не стане.

Другото което трябва да знаете, е, че има случаи в които проблема не е застъпен в помощната информация. Тогава трябва да отидете на сайта на производителя, и в раздела за поддръжка да потърсите решението. Там никой няма да ви изяде. На тях дори им изнася да ви помагат, защото разчитат като ти кажат как да си решите проблема, с тяхна помощ, и после да кажете, ей то работи, имах проблем, питах производителя, производителя ми каза направи това и това, аз го направих и проблема се реши.

Същото важи и за документацията на вашата операционна система. Четейки документацията на операционната система научавате как работи Операционната Система, какви ресурси ползва кога какво взема като ресурс, кога какво освобождава като ресурс, съпоставяйки информацията, в документацията на операционната система и на антивирусната програма, можете с точност сами да откриете къде точно е проблема и как да го коригирате.

Това са важни неща, които трябва да знаете, така опознавате хем вашата операционна система и хем вашата антивирусната програма и техните възможности, и така се научавате да ги ползвате ефективно

За повече детайли свързани конкретно с комбинацията между версия на Windows и версия на антивирусна програма прочетете документацията на Windows и антивирусната програма.

Преди да питате по форумите, уверете се че сте прочели документацията на операционната система и на антивирусната програма. ако там няма резултат, уверете се че сте търсили помощ от поддръжката на производителите на операционната система и антивирусната програма, и ако те не са могли да ви помогнат, чак тогава, питайте по форумите.



Хората ви поддържат бази данни с въпроси и отговори, имат експертен персонал за консултации, прилагат ви документи и помощна информация и всичко това, е там за да го ползвате като помощно средство да си решите проблема. то не е украшение, не е случайност, то е там за да ви помага.

**Автор:** Кристиан Александров

# Безвъзвратно триене

Undelete е нож с две остриета, както може да ни спаси файловете, така може и да ни разобличи. В тази статия ще гледаме на Undelete като нещото което е пречка за нас и трябва да го елиминираме на всяка цена.

Представете си що за позор, хакнали сте и сте изтеглили нещо от РС на жертвата, и ченгетата ви арестуват, ровят ви по диска и ужким изтритото нещо е там – позор, полицаите намират търсеното доказателство. Това не бива да до допускаме.

Сега да започнем с това какво представлява триенето, защо е то не е сигурно триене, и как ще ви извадят данните от диска. Ако се стигне до там да го отворят, и с хардуерни методи в лаборатории полицията да ви вадят данните – приемаме че ще го правят без да им мигне окото. Това също не бива да го допускаме.

При самото триене, натискане на клавиша делете – файла се мести в кошчето. При изпразване на кошчето – файлът ужким се изтрива. Това е най-несигурният възможен начин, на триене, тъй като следи от файла остават на 2 места – там където файлът си е бил по начало и там дето е кошчето. Ако триете директно с прескачане на кошчето – шифт+делете, оставяте една следа само там където е бил файла. Достатъчно несигурно за да ви извадят файла изпод носа.

При тези видове триене, файлът е там, ОС маркира секторите като празни въпреки че файлът е там и оставяйки записа на файла във файловата таблица почти непокътнат, само първият символ от името на файла е изтрит, и записа не може да бъде достъпен от ОС и програми. Така тя сама заблуждава себе си и програмите, че там наистина е празно и при следващото писане, тя презаписва секторите, с друг файл, което на теория прави файла невъзстановим за софтуерните методи за възстановяване, но на практика има една особеност, на която никой не обръща внимание и не и отдава значение, а точно тази особеност е важна и позволява да се възстановяват части от файлове, въпреки че е "презаписано". Тази особеност е Slack space – пусто място. Това място, се появява когато един файл или парче от файла не може да запълни целия клъстер. Това място е мястото от края на файла до края на клъстера, или от края на парчето на файла до края на клъстера. Това място е много коварно, от него се вадят много данни, но основно непълни файлове и непълни парчета. Пак стигат като количества за да ви разобличат. Как става така? Лесно, примерно, в 4 килобайтов клъстер на хард диска, (клъстера на нтфс е с големина 4 килобайта по подразбиране) има един 4 килобайтов файл, като изтриете този файл и на негово място запишете файл с обем 1 килобайт, то ще останат 3 килобайта пусто място (slack space) само първият килобайт ще е презаписан, а останалите 3 остават непокътнати. Ако някой реши да търси данни в тези 3 непокътнати килобайта пусто място, ще намери 75% от данните на стария файл който е бил ужким изтрит. За това трябва и slack space да се прочисти. Колко точно е slack space зависи от обема на клъстера и обема на файла който се намира в клъстера, или от обема на парчето от файла което се намира в клъстера.

С хардуерни методи и изследване на профилите на магнитните полета – аномалията на магнитните полета (обяснено е по-нататък в статията) може да се извади файла, или парчета от него с различна степен на изкривяване на данните. Изкривяването зависи от множество фактори – лоши сектори, степен на съвършеност и прецизност на метода който се използва, съвършеност, чувствителност и прецизност на апаратурата, оборудване, изолация и чистота на помещението в което се възстановява, защита от външни магнитни полета които могат да попречат на апаратурата или повлияят на магнитните плочи на диска и полетата, грамотност и способности на възстановяващият и колко пъти местата са презаписани. Част от изкривяванията могат да се възстановят други не.

Софтуерно възстановяване на файлове – става със специални програми – recover my files, easy recovery и други програми като тези. Тези програми работят общо казано в два режима:

**UNDELETE** – търси се изтритият файл, като го намери, секторите се маркират за заети и записва във файловата таблица се коригира. Така файлът е отново видим и достъпен.

**RAW READING** – последователно четене на всички клъстери и сектори в клъстера, и опит за разпознаване на намереното. Този вид търсене няма нужда от данни във файловата таблица. Този начин работи по-добре при редовно поддръжан диск, защото програмата няма да се чуди кое парче от кой файл е и къде във файла е парчето. Направо прочита секторите, събира информацията като един файл, създава нов файл на мястото на изтрития с намереното съдържание и информация за файла, създава нови записи за новият файл във файловата таблица, така че да е видим, за ОС и програмите. Ако диска не е редовно поддръжан, възстановяването при RAW четене, ще изпитва сериозни затруднения при разпознаването на частите – коя част към кой файл принадлежи и къде точно във файла се намира тази част. Възможно е дори да ви направи един огромен файл, с всичкото съдържание, което е видяно и точно в този ред в, който е видяно. Възможно е по-грамотните алгоритми за разпознаване на парчетата, и определяне на файла към който принадлежат, и точното място на парчето във файла, да дадат по-добри резултати, но основно зависи от това какво точно е предизвикало "изчезването" на файла, и колко редовно е бил поддържан диска. Този вид възстановяване, обикновено, работи само най-яркия слой на магнитните полета в аномалията на магнитните полета тъй като за да се избегнат грешки при четене, още на ниво БИОС на хард диска, самият хард диск е инструктиран, използвайки главите си за четене, да чете само най-яркия слой и да пренебрегват всички други.

Всеки твърд диск или дискета си има едно основно магнитно поле, то е най-ярко и най-мощно, и хард диска чете само това поле и пренебрегва всички останали. Аномалията на магнитните полета спомената по-горе се предизвиква от това че на хард дисковете, при правенето на диска когато върху плочите производителя като поставя магнитното покритие то се състои от няколко слоя, което прави така, че всеки слой на магнитното покритие, има свое магнитно поле. Тези магнитни полета резонансно си въздействат едно на друго като резултат, от което взаимно въздействие, като резултат се появяват и резонансни магнитни полета. За да се избегне объркването в тая какофония от полета, още на ниво БИОС на хард диска, самият хард диск е инструктиран, използвайки главите си за четене, да чете само най-яркото поле – слой (ще го наричам основно поле) и да пренебрегват всички други. Как диска разбира кое поле е основното, и кое поле е резонансно – това е работа на биос и електрониката на диска. Способността на диска да се ориентира бързо в тая какофония от полета определя и неговата скорост.

Ако производителят на диска пише че достъп до данни се осъществява за 9 милисекунди, то за тези 9 милисекунди диска трябва да свърши страшно много работа – да провери във файловата таблица, къде е информацията за файла и да я намери, да прочете от файловата таблица на коя пътека и в кой клъстер е файла или парчето то файла, да си намери главите на правилната пътека, да изчака сектора да мине под главите, да прочете тая какофония от по-

лета, над сектора, от тях да отсее само основното поле от всички останали полета, и да види какви екстремуми има в това поле, за да декодира данните – реда на екстремумите представлява данните кодирани на диска, да се увери че данните са прочетени без грешка, ако има грешка да коригира грешката и да ги прати на контролера. Това е причината диска да е най-бавната част в един компютър.

Екстремум узнава връхна или низна точка в магнитно поле – ако връхната точка представлява двоична единица, то низната точка е двоична нула. Тяхното редуване представлява кодирането на данните в хард диска.

След като разбрахме какво става под капака на диска, трябва да споменем че с това не се изчерпва аномалията на магнитните полета. Различните модели, според технологията на производство и показатели имат от 5 до 8 такива полета, част от тях основни полета от слоевете на магнитното покритие, другата част, резонансно получени от взаимодействието между основните полета. Данни може да има във всеки от тези слоеве, и хардуерните методи, които подробно изследват магнитното поле, реално изследват всички полетата в тази аномалия, за нещо което прилича на данни, и като намерят такова нещо дето прилича на данни се опитват да видят какво има там или е грешка. Заради това винаги има изкривявания по данните при хардуерно възстановяване.

При запис на данните върху магнитното покритие, се променя и най-яркото от магнитните полета, тъй като то е най-мощното, то въздейства на останалите, като на някои въздейства повече на други по-малко, като налага своят профил на останалите. При изтриване на файла, слойът остава, защото магнитното покритие е частично променено (премахнат е маркера зает клъстер и е поставен маркер свободен клъстер), но важното което ни интересува е непокътното – кодировката на данните е същата и остава, нали секторите се маркират празни въпреки това че има нещо. Това позволява на профила с времето да се "просмуче" в другите полета. Това става за части от секундата, така че само докато помислите полето вече се е "просмукало". Разбира се, колкото повече се "просмуква" едно поле то толкова повече отслабва. За това според това какво се записва, основното поле е различно мощно и се "просмуква" различно в дълбочина на слоевете. Това е общо казано аномалията на магнитните полета.

За да се предпазите, от това, някой да не ви извади данните от тези полета, и от основното поле, и от останалите полета, всичките полета трябва да получат някакъв безсмислен профил, примерно само нули или само единици, или безразборно подбрани какви да е символи без каквато и да била връзка между символите. Това става със специални програми за почистване на диска частично скрубване и пълно скрубване. Тези програми работят на принципа записват въпросният низ от символи, после го изтриват, после пак го записват после пак го изтриват, и така много пъти, 12 – 15 пъти са достатъчни да спрат каквото и да било като възстановяване, независимо хардуерно или софтуерно и независимо колко е свършено като метод и технология и колко грамотен, способен, ловък и умел е човека който се е захванал със задачата. Това се прави с цел полето винаги да се достатъчно силно за да се "просмуче" във всички слоеве, и всички слоеве да имат еднакъв профил.

При пълното скрубване, целия диск всички клъстери и сектори биват така многократно презаписвани, безвъзвратно се унищожават всички файлове, дялове, файловата таблица, таблицата с дялове, всичката информация от нулевата пътека – абсолютно всичко от диска бива безвъзвратно изтрито. Добре е да се прави като ще продавате диска, или ще го подарявате или ако се отървявате по някакъв начин и по някаква причина от него. След такава намеса ще трябва да създадете дяловете, да ги форматирате, и чак тогава ще можете да ползвате диска. Програми които правят това са програми от сорта на Wipedrive, drivescrubber.

При тези многократни презаписвания, ще се стигна до ситуация в която програмите за възс-

тановяване ще виждат някакви файлове, основно текстови с различни размери и безсмислено съдържание, тъй като RAW четенето ще е силно объркано от това което намира и няма да може да определи, кое какво е и къде му е мястото, и какво е било истинското съдържание или въобще да не намерят нищо – нали това е нашата цел.

При непълното скрубване, задачата е по-сложна, трябва да се изпълнят повече неща. Първо програмата трябва да прегледа диска и да установи кои клъстери и сектори в клъстера са заети и празни. После от заетите трябва да установи, кои са за скрубване и кои не са за скрубване. За да определи това, програмата трябва да прегледа файловата таблица, да види, кои записи са валидни, и кои не. Чрез файловата таблица програмата решава кои от заетите сектори ще скрубва защото съдържат изтрети данни и кои няма да пипа, защото имат файлове които са видими за ОС и програмите и потребителя може да ги ползва, да се определи свободното място на диска/дяла, След като всичко това се определи, програмата трябва да определи коя част от клъстера, да се приеме за slack space и тя също да се изчисти. Не всички програми почистват и slack space, така че при избор на програма, търсете такава, която чисти и slack space. Чак след като всички проверки са извършени, по диска, файловата таблица и файловата система, slack space, свободно място.... програмата слага всичката тази информация в рам, а не на диска, и почва да трие и презаписва, там където програмата е решила че ще трие.

При самото триене, преди всичко, познайте – изпразва се кошчето, после от файловата таблица, се премахват всички невалидни по някаква причина записи – триене, забил компютър по време на писане, повреден от бъгава програма, вирус, червей, троянски кон, мрежов експлойт, бедствие, нестабилен ток, токови удари, удари причинили физически лоши сектори, логически лоши сектори, магнити близо или върху хард диска – всякакви причини които се сетите. След това от диска се премахват всички файлове и парчета от файлове останали без описание във файловата таблица, и slack space – мястото което остава между края на файла или парчето от файла и края на клъстера, заедно с останалото празно място, се презаписва, многократно пише безсмислени низове от символи, после ги трие, после пак – и така 12 – 15 пъти са достатъчни да спрат каквото и да било като възстановяване, независимо хардуерно или софтуерно и независимо колко е свършено като метод и технология и колко грамотен, способен, ловък и умел е човека който се е захванал със задачата. Пак по същият начин за да може профила на безсмислените символи да се просмуче в полетата от магнитната аномалия и в слоевете съставлящи магнитното покритие на хард диска, за да може пак всичките да имат еднороден профил – на безсмислени глупости, за да няма възстановяване, независимо от условията и методите на възстановяване. При тези многократни презаписвания, ще се стигна до ситуация в която програмите за възстановяване ще виждат някакви файлове, основно текстови с различни размери и безсмислено съдържание, тъй като RAW четенето ще е силно объркано от това което намира и няма да може да определи, кое какво е и къде му е мястото, и какво е било истинското съдържание – нали това е нашата цел. Програми които правят това са програми от сорта на evidence eliminator, system shield, secure clean. Правете това редовно – доволно често примерно през нощ или всяка нощ, докато спите не използвате компютъра, защо не го оставите да се почиства докато спите? Това е дълга процедура, изисква много време, за са се завърши и изпълни добре, и ако го правите през времето когато спите, това няма да ви пречи. Вие тогава не използвате компютъра и тази активност, няма да ви пречи.

Ще свършите една полезна работа. На сутринта компютъра ще е почистен от следите. Ще имате чист диск, а и за това, че диска ще се чисти редовно, ще имате предимството, че диска, ще намира файловете по-бързо, защото махайки излишната информация и боклуците с изтретите файлове от файловата таблица, вие улеснявате диска, по-малка таблица трябва да бъде преровена, щом е по-малка таблицата, значи за по-малко време ще се прерови, и въпросният файл ще бъде открит по-бързо.



Що се отнася за външните памети като USB преносими носители, флаш и секюре карти за фотоапарати и камери, memory stick и прочието такива външни памети за файлове при тези устройства не съм запознат как точно става записа и четенето на файловете, магнитно-оптични устройства, не съм и запознат как става триенето и как, къде и какви следи остават, знам само че и при тях остават следи, които трябва да се заличават.

Що се отнася до ленти, zip/jaz/jar устройства, дискети, външни твърди дискове (такива които се включват в сериен, паралелен, усб или firewire порт, инфрачервен и блутуут а не се включват на IDE/SCSI контролер) при тях, това как се пише, как се чете, как се трие и къде и какви следи остават, и как се търсят тия следи, как се възстановява е както при обикновените твърди дискове, за които е и главната част от статията. Има подобна програма за прочистване, на последните 2 групи устройства, само че тя прави само пълно прочистване на цялото устройство и заличава всички данни безвъзвратно. Програмата се казва media wiper.

**Автор:** Кристиан Александров

## PHP на домашното PC

Напоследък все повече хора се интересуват от станалия изключително популярен език за уеб програмиране PHP. Без съмнение това е най-разпространеният server-side скриптов език, като безспорните му предимства пред останалите езици от неговия вид са известни на уеб разработчиците - той е лесен за усвояване и използване, проектиран е специално за интернет, има изключително добра съвместимост с най-използваните RDBMS (системи за управление на релационни бази данни), и най-важното - всичкият необходим софтуер за използването му е open source, т.е. е напълно базплатен.

Проблемът при тепърва откриващите PHP света уеб дизайнери/разработчици се състои най-често в инсталирането и конфигурирането на среда за изпълнение на PHP скриптове. Въпросът "Как да стартирам/да подкарам/да видя скрипт на PHP" е един от най-често срещаните при начинаещите. Причината за объркването е различния механизъм по който действа PHP спрямо HTML - единият език се изпълнява на уеб сървър, а другият на браузъра на потребителя, разглеждащ страницата. Тази статия цели да разясни какво е нужно за да накарате домашната си машина да обработва PHP скриптове и защо въобще е необходим допълнителен софтуер за това.

Принципно има няколко критерия за разделянето на различните програмни езици в групи. Един от тях е типа транслиране, което се осъществява за даден език. За да може една програма да бъде изпълнявана от процесора на компютъра, тя трябва да бъде на машинен език (т.е. кода ѝ да е съставен от единици и нули подредени в определен логически ред). Съвременните езици за програмиране обаче са от т.нар. "високо ниво" - те наподобяват човешките езици и са проектирани така, че кода на програмите да се разбира от хората, а не от компютрите. За да работят програмите, те трябва да бъдат преведени на машинен език с помощта на транслатори. Тези транслатори биват два вида - компилатори и интерпретатори. Компилаторите превеждат целия код и след компилирането му създават изпълними файлове (\*.exe, \*.com), докато интерпретаторите превеждат кода директно по време на самото изпълнение на програмата. Така няма нужда програмата да бъде предварително компилирана. Примери за езици, които се компилират, са Basic, Pascal, C/C++ и други, а за езици, които се интерпретират - Java, Python, както и повечето уеб езици.

PHP спада към езиците, които се интерпретират. Това налага използването на PHP интерпретатор при всяко стартиране на скрипт. Повечето начинаещи се опитват да стартират такива скриптове чрез браузъра си - запомнете, това е НЕВЪЗМОЖНО! Колкото и да опитвате това, винаги ще виждате само кода на скрипта, без той да бива реално изпълнен. Всъщност браузърите също могат да бъдат разглеждани като интерпретатори, но за ограничен брой езици - HTML, CSS, XML, XSL, JavaScript, VBScript - и толкова. Когато някой потребител отваря даде-

на уеб страница, той на практика сваля (download-ва тук е най-точната дума) кода на страницата, след което интерпретатора на браузъра изпълнява поставените му задачи. Този подход обаче прави уеб сайтовете малко или много статични - съвременните сайтове разчитат на динамично съдържание, те са интерактивни - потребителите имат много по-голяма свобода за различни действия, като например да публикуват сами определено съдържание или да избират сами какво от сайтовете да виждат и какво не. Ясно е, че обикновения подход (изпълнение на HTML страници на браузъра) не може да осигури това - нужно е друго решение. Тук на помощ идват сървърните езици и базите данни. Механизмът е прост - потребителят извършва някакво действие на страницата, изпратената информация се обработва на сървъра, съдържанието на базата данни се актуализира и след това всеки друг потребител може да види новото съдържание. Обработката на сървъра се осъществява именно от езици като PHP. Поради тази причина не е нужно браузърите да имат вградени интерпретатори за PHP - по този начин се изгубва всякакъв смисъл (динамичното съдържание вече не е възможно), още повече че подобни езици се използват в интернет от години - такива са например JavaScript и VBScript.

Въпросът е - какво е нужно на един уебмастър за да може да изпълнява PHP скриптове на своята машина? Отговорът - нужен е уеб сървър, PHP интерпретатор и в повечето случаи система за управление на бази данни (най-често това е MySQL). Поради голямото разнообразие на операционни системи и сървъри, тази статия не може да даде универсално упътване за това как да инсталирате всички необходим софтуер на вашият компютър. Най-добре е да посетите последователно следните адреси, да изтеглите съответстващите на вашата операционна система версии на продуктите и да ги конфигурирате сами следвайки инструкциите на инсталацията:

**<http://www.apache.org/>**

**<http://www.php.net/>**

**<http://www.mysql.com/>**

На първия адрес ще намерите най-широко използвания уеб сървър в света - Apache. Той е напълно безплатен и има версии за повечето операционни системи. Разбира се има и други алтернативи за уеб сървър - PWS 3 и 4 или по-нов, IIS 3 и 4 или по-нов, OmniHTTPd 2.0b1 или по-нов, Oreilly Website Pro, Xitami, Netscape Enterprise Server и други. От втория сайт можете да си свалите интерпретатора за PHP, който също е open source. Ако използвате готови скриптове, вероятността те да използват база данни е голяма - в такъв случай посетете третия сайт. Там ще намерите една от най-бързите и надеждни системи за управление на бази данни - MySQL. Другите възможности за такива системи са PostgreSQL, mSQL, MSSQL, Interbase и други. Моята препоръка е обаче да започнете първо с MySQL, най-вече поради факта, че съществуват хиляди готови безплатни скриптове, които работят именно с MySQL - от тях можете да научите много. Следвайте внимателно инструкциите за инсталация и за нула време компютърът ви ще се превърне в работещ уеб сървър.

Разбира се, отделното инсталиране на сървър, интерпретатор и бази данни не винаги е много лесно нещо. Например операционни системи като Windows 95/98/Me по принцип не са предназначени за сървъри. За щастие съществуват достатъчно добри инсталатори, които не само инсталират отделните софтуерни пакети, но и ги конфигурират за съвместна работа. Чрез тях можете да си спестите доста трудности при конфигурирането на сървър в домашни условия. Три от най-известните решения в тази насока са FoxServ (личния ми фаворит), PHPTriad и OmniHTTPd. Ще разгледаме трите програми поотделно.

**FoxServ** [Download - 28 MB] - Това е може би най-добрият пакет от трите. Единственото неудобство при него е може би доста големия инсталационен файл. За сметка на това освен стандартната комбинация на сървър, PHP интерпретатор и база данни инсталацията включва и два удобни инструмента за работа с MySQL и оптимизиращия енджин на Zend. Версиите на отделните компоненти са по-нови от тези при другите инсталатори. Допълнително удобство е Control Panel-а, чрез който много бързо можете да управлявате сървъра си. Също така получавате и подробна документация за Apache сървъра си (в .pdf и .html формат) и MySQL базите си данни (.pdf). FoxServ 3.1b включва:

- **Apache 1.3.26** - **PHP 4.2.1** - **Perl 5.6.1.632** - **MySQL 3.23.51** - **phpMyAdmin 2.2.6**  
- **MySQL Front 2.40** - **Zend Optimizer 2.0.2**

**PHPTriad** [Download - 13 MB] - Има поддръжка само за Windows, като тук инсталацията е сведена до натискане на един бутон. Изключително удобен и приятен инсталатор - ако сега започвате да се занимавате с PHP, той ще ви даде всичко от което се нуждаете. Има и удобен Control Panel, чрез който ще можете бързо да управлявате вашия сървър. За съжаление този продукт вече е платен и е с ново име - Sokkit. Последната излязла версия на безплатния инсталатор съдържа вече остарели версии на отделните компоненти (за пример текущата версия на PHP към момента на писане на статията е 4.3.1, докато в пакета е включена версия 4.1.1). Все пак ако желаете просто да се учите на PHP, този инсталатор ще ви свърши чудесна работа. Допълнителен плюс е, че файлът е наполовина по-малък от този на FoxServ.

PHPTriad 2.2 включва:

- **Apache 1.3.23** - **PHP 4.1.1** - **MySQL 3.23.48** - **phpMyAdmin 2.2.3**

**OmniHTTPd** [Download - 2.1 MB] - Най-малката програма. С нея получавате сървър OmniHTTPd с поддръжка за PHP 4.2.2, но без MySQL и другите екстри на изброените по-горе инсталатори. Идеална за тези които искат да учат PHP като започнат от най-елементарното и малко по малко напреднат с по-сложните неща. Не се препоръчва ако ще ползвате готови скриптове, използващи бази данни.

Единствения проблем на подобни инсталатори е, че в повечето случаи те не включват най-новите версии на отделните продукти поради простата причина, че последните биват актуализирани доста често.

След като сте си избрали инсталационен пакет (или пък сте инсталирали и конфигурирали всичко сами), ви остава само да изпробвате дали всичко работи нормално. Преместете файловете с PHP скриптовите в директорията на сървъра, предназначена за сайтовете (най-често това е C:\Apache\htdocs, ако работите с Windows и Apache). Стартирайте сървъра (C:\Apache\Apache.exe) и ще се появи DOS-овски прозорец с подобен текст в него - "Apache/1.3.23 (Win32) running...". Ако работите с Linux или друга ОС стартирането ще е подобно. Когато вече и сървъра работи, ви остава просто да отворите страниците, които желаете, чрез написване в полето за адреси на браузъра си следното:

**http://localhost/imenafaila.php** или **http://127.0.0.1/imenafaila.php**

Забележете, че не браузъра ви изпълнява скриптовите - това се върши от вградения в сървъра интерпретатор за тях. Също така съм длъжен да отбележа, че е много препоръчително да използвате инсталирания уеб сървър на вашата машината CAMO за тестване, експериментирание и практическо усвояване на уроци за PHP и MySQL. Превръщането на домашното ви PC в истински уеб сървър НЕ Е ПРЕПОРЪЧИТЕЛНО при никакви обстоятелства. От една страна сигурността е на много ниско ниво (особено ако работите с Windows), от друга производителността и скоростта на подобен сървър биха били доста ниски в сравнение със тези на истинските, професионални сървъри, които имат най-често гигабайти RAM и при които връзката се измерва в MB/s...

Това е - надявам се с това да ви се е изяснило как се изпълняват PHP скриптове и какво ви е нужно за това. **Успех!**

**Подготвил:** The ONE **Източник:** [www.bestof.hit.bg](http://www.bestof.hit.bg)

## JavaScript - Математически операции

Можете да извършвате различни математически операции с променливите от целочислен тип, както и с променливите с плаваща запетая. Операторите са следните:

+ **събиране**    - **изваждане**    \* **умножение**    / **деление**    % **целочислено деление**

Когато изпълнявате математическа операция с повече числа, можете да използвате скоби за да разграничите приоритетите на изчисленията. Например ако искате да съберете 2 със 5, а после да разделите резултата на три трябва да напишете  $(2+5)/3$ . Ако изпуснете скобите ще се получи  $2+5/3$ . Тогава първо 5 ще се раздели на 3, а после към резултата ще се прибави 2. Запомнете, че умножението и делението са с по-голям приоритет от събирането и изваждането.

Нека да направим един скрипт в който въвеждате годината в която сте родени, след това да изчислява на колко години ще бъдете през 2010 година и най-накрая да се отваря прозорец, който да съобщава резултата:

```
<HTML>
<HEAD>
<TITLE>javascript</TITLE>
<HEAD>
<BODY>
<SCRIPT LANGUAGE="javascript">
<!--
var year=prompt("Въведете година на раждане","1980")
old=2010-year
alert("През 2010 година Вие ще бъдете на "+old+" години")
//-->
</SCRIPT>
</BODY>
</HTML>
```

var е друг начин да се обявяват променливите (var name="Ivan" , var number=14). Със prompt се отваря прозорец с поле, в което можете да въвеждате информация. Структурата на prompt е следната - prompt("message","value") където message е съобщението което ще се появи заедно с прозореца, а value е някаква начална стойност. Можете да пропуснете value, тогава полето ще бъде празно. В примера чрез prompt присвояваме на променливата year стойността която ще напишете в полето. В следващия ред създаваме променлива old на която присвояваме стойност 2010 минус стойността на year. Ако оставите в полето по подразбиране числото 1980, то тогава year ще бъде равно на 1980, а old на 30 (2010-1980). И най-накрая извеждаме съобщение, че през 2010 г. вие ще бъдете на old години, като променливата old се заменя от стойността ѝ. Самото съобщение се извежда с alert прозорец (alert ("През 2010 година Вие ще бъдете на "+old+" години")).

Този примерен скрипт не прави проверка за верността на въведените от потребителя данни, така че ако въведете някакъв стринг вместо число, скрипта ще даде грешка.

Други аритметични операции с променливи са събиране с едно (x++) и изваждане с едно (x--). x++ е същото като x=x+1, а x-- като x=x-1. Ако например променливата x има стойност 5, след x++ x ще е равно на 6.

Ако присвоявате стойността на x на друга променлива, например y и в същото време увеличите x с едно, стойността на двете променливи ще бъде една и съща.

```
y=x++
```

..така x и y ще бъдат с еднаква стойност. Ако напишете обаче:

```
y=++x
```

.. y ще приеме стойността на x и чак след това x ще се увеличи с 1. Значи да приемем че x=10. Тогава при y=x++ x ще стане 11 и y ще стане 10. Ако напишем y=++x тогава y ще стане 11, а x 11.

**Подготвил:** The ONE **Източник:** [www.bestof.hit.bg](http://www.bestof.hit.bg)